Patent Application
Docket No. TI-29645

CERTIFICATE OF MAILING BY EXPRESS MAIL

"EXPRESS MAIL" Mailing Label No. EJ042687290US
Date of Deposit 8-14-00
I hereby certify that this paper or fee is being deposited with the U.S. Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231.

Type or Print Name: Joy L. Mowery

Signature

HIGH-SPEED, HIGH GRANULARITY BAUD CLOCK GENERATION

FIELD OF THE INVENTION

The invention relates generally to baud clock generation and baud rate detection for serial communications and, more particularly, to baud clock generation and baud rate detection for serial communication interfaces that use oversampling and a fixed frequency clock.

BACKGROUND OF THE INVENTION

In conventional serial communication interfaces that use oversampling during data reception (also referred to herein as oversampling serial communication interfaces), for example universal asynchronous receiver/transmitters (UARTs), the baud rate is typically obtained by dividing the base clock rate of the interface by a number that is an integer

multiple of the oversampling factor. Thus, the possible baud rates are given by: base clock frequency/((oversampling factor) x (integer divisor)), where the oversampling factor is expressed, for example, in baud clock pulses (i.e., samples) per bit. If the base clock of the serial communication interface is set at a fixed frequency, as is typical, then there are significant gaps between the possible baud rates. For example, for a base clock frequency of 1.8432 MHZ and an oversampling factor of 16 pulses/bit, the baud clock rates produced by using integer divisors of 1, 2 and 3 are 115,200 baud (bits/second in this example), 57,600 baud and 38,400 baud, respectively. Note the significant gap between the adjacent baud rates of 115,200 and 57,600. In this situation, if there is a need for a baud rate that does not fall near any of the available baud rates, for example 76,800, it is typically necessary to provide a higher base clock frequency, from which an acceptable approximation of the desired baud rate can be derived. However, the increased base clock frequency disadvantageously requires an increase in the power used by the serial communication interface.

It is therefore desirable to increase the number of available baud rates without increasing the base clock frequency of the serial communication interface.

The present invention provides for generation of a baud clock by dividing the base clock of the serial communication interface by one of a plurality of possible composite divisors. Each composite divisor is indicative of a minimum time interval between adjacent

pulses of the associated baud clock, and further indicates that at least one pair of adjacent pulses within each symbol interval of the associated baud clock are to be separated by an extended time interval which is longer than the minimum time interval. The composite divisor advantageously results in a large number of available baud clock rates without increasing the base clock frequency.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGURE 1 diagrammatically illustrates an exemplary embodiment of a baud clock generator according to the present invention.

FIGURE 2 diagrammatically illustrates an exemplary embodiment of the divisor generator of FIGURE 1.

FIGURE 3 diagrammatically illustrates an exemplary embodiment of the clock divider of FIGURE 1.

FIGURE 4 illustrates exemplary contents of the pulse pattern memory of FIGURE 3.

FIGURE 5 illustrates exemplary operations which can be performed by the clock divider embodiment of FIGURE 3.

FIGURE 6 is a timing diagram which illustrates an example of a baud clock signal generated by the clock divider of FIGURE 3.

FIGURE 7 diagrammatically illustrates another exemplary embodiment of the divisor generator of FIGURE 1.

FIGURE 8 illustrates a portion of the contents of an exemplary embodiment of the look-up table of FIGURE 7.

FIGURE 9 diagrammatically illustrates pertinent portions of an exemplary embodiment of a data processing and communication system according to the invention.

FIGURE 10 illustrates exemplary operations which can be performed by the embodiments of FIGURES 1-9.

FIGURE 11 illustrates an exemplary embodiment of a baud rate detector according to the invention.

DETAILED DESCRIPTION

FIGURE 1 diagrammatically illustrates an exemplary embodiment of a baud clock generator according to the invention. The baud clock generator can be provided, for example, in an oversampling serial communication interface such as a UART. In FIGURE 1, information indicative of the desired baud rate, for example in units of bits/second, is input to a divisor generator 11. In response to the desired baud rate, the divisor generator 11 produces and outputs an enhanced granularity divisor D_{EG}. This enhanced granularity divisor is a composite divisor including a first divisor component D and a second divisor component REMD. These divisor components are input to a clock divider 13 which, in response to the first and second divisor components, divides the base clock signal of the oversampling serial communication interface to produce a baud clock signal that approximates the desired baud rate. The baud clock signal 15 is provided to a sampler of the oversampling serial communication interface, which can use the baud clock signal in conventional fashion to sample serial communication data received at the interface. The baud clock signal 15 can also be used in conventional fashion to transmit data from the interface.

The first divisor component D corresponds to the aforementioned integer divisor that is conventionally used to generate baud rates. Using the conventional technique, the integer divisor that will produce the closest approximation to a desired baud rate is given by the

quotient of the following division operation: base clock frequency/(K x desired baud rate), where K is the oversampling factor associated with the oversampling serial communication interface. Advantageously, the invention utilizes also the modulo K remainder produced by this division operation. By using the modulo K remainder, the invention produces K-1 intermediate baud rates between any given pair of adjacent baud rates that would be produced by the conventional technique. The second divisor component REMD of FIGURE 1 corresponds to this modulo K remainder.

FIGURE 2 diagrammatically illustrates an exemplary embodiment of the divisor generator of FIGURE 1. In the embodiment of FIGURE 2, a multiplier 31 multiplies the desired baud rate by the oversampling factor K. A divider 35 divides the base clock rate of the serial communication interface by the output 33 of the multiplier 31, namely K x (desired baud rate), producing a quotient and a modulo K remainder. The modulo K remainder is the second divisor component REMD of FIGURE 1, and the quotient is the first divisor component D of FIGURE 1.

FIGURE 3 diagrammatically illustrates an exemplary embodiment of the clock divider 13 of FIGURE 1. In this embodiment, the base clock of the serial communication interface is applied to the count input of a base clock counter 51 which counts pulses of the base clock. After a predetermined number of base clock pulses, the output 58 of counter 51

triggers a pulse generator 52 to output a baud clock pulse. The counter 51 includes a load select input 52 that determines which of two possible target count values will be loaded into the counter 51. More specifically, the counter 51 is selectively loaded with a target count value of D or D + 1 in response to input 52. Thus, when D is selected as the target count value, the counter 51 triggers a baud clock pulse for every D pulses of the base clock. Similarly, when D + 1 is selected as the target count value, the counter 51 triggers a baud clock pulse for every D + 1 pulses of the base clock.

The load select input 52 is driven from a pulse pattern memory 54 that includes K entries, one for each possible value of the modulo K remainder REMD. Each of the K entries in the pulse pattern memory 54 includes K bits having a value of either 1 or 0. The second divisor component REMD is input to the pulse pattern memory 54, and selects the corresponding one of the K entries. The pulse pattern memory 54 has a pointer input 56 which points to one of the K bits of the entry selected by REMD. The pointer value at 56 is provided by a baud clock counter 53 whose count input is coupled to the baud clock 15. The counter 53 counts cyclically through K states, thereby repeatedly tracking the K baud clock pulses used to sample each bit received at the serial communication interface. The output count value of counter 53 is provided as the pointer value at 56. Thus, for each baud clock pulse within a given bit interval of the serial communication interface, the pointer 56 points

to a specific one of the K bit locations in the pulse pattern memory entry selected by REMD. If the bit value pointed to is 0, then the pulse pattern memory 54 outputs a load select value of 0, thereby selecting D to be loaded as the count target for counter 51. If the bit value pointed to is 1, then the pulse pattern memory 54 outputs a load select value of 1, thereby selecting D + 1 to be loaded as the count target for counter 51.

FIGURE 4 illustrates exemplary contents of the pulse pattern memory 54 of FIGURE 3. In this example, the oversampling factor K is 16, so there are 16 entries in the pulse position memory, each entry having 16 bits. For this oversampling factor of K=16, and recalling that REMD is a modulo K remainder, there are 16 possible values of REMD, namely 0-15. As shown in FIGURE 4, the entry corresponding to REMD=0 includes all 0s. In this instance, all pulses of the baud clock will be separated from one another by a time interval corresponding to D pulses of the base clock, just as in the prior art.

As shown by the example of FIGURES 3 and 4, a non-zero value of REMD specifies a number of pairs of adjacent pulses within each bit interval of the baud clock 15 will be separated by a time interval corresponding to D+1 pulses of the base clock. That is, for example, when REMD=1, the corresponding entry in the pulse position memory has one bit with a value of 1, so there will be a time interval corresponding to D+1 base clock pulses between 1 pair of adjacent pulses in each bit interval of the baud clock. This time interval

corresponding to D+1 base clock pulses is longer than the aforementioned time interval associated with D base clock pulses and, in this example, will occur between the third and fourth baud clock pulses of each bit interval, because the fourth bit in the pulse pattern memory entry corresponding to REMD=1 has a value of 1. Similarly, for a value of REMD=2, two pairs of adjacent baud clock pulses in each bit interval will be separated by the longer (D+1) time interval, namely the third and fourth pulses and the eleventh and twelfth pulses. As can be seen in the example of FIGURE 4, it can be advantageous to distribute the 1s approximately evenly within the respective pulse position memory entries, thus approximately evenly distributing the longer time intervals throughout the bit interval.

FIGURE 5 illustrates exemplary operations which can be performed by the clock divisor of embodiment of FIGURE 3. After initializing base clock and baud clock count values at 80 and 81, respectively, the first base clock pulse is received at 82, causing the base clock count to increment by one. It is thereafter determined at 83 whether the pulse pattern memory bit selected by REMD and the baud clock count value is 1 or 0. If the selected bit is 0, then D base clock pulses will be counted, as illustrated at 84. If the selected bit is 1, then D + 1 base clock pulses will be counted, as illustrated at 85. After the selected number (D or D+1) of base clock pulses have been counted at 84 or 85, a baud clock pulse is generated at 86. The baud clock pulse at 86 causes the baud clock count value to increment

by one as shown at 87. The aforementioned steps at 81-87 are then repeated until the end of the current bit interval is reached at 88, namely when the baud clock count value reaches K. At this point, the count values are reset again at 80 and 81.

FIGURE 6 is a timing diagram which illustrates an example of the baud clock 15 produced by the clock divider embodiment of FIGURE 3. In the example of FIGURE 6, the oversampling factor K is 16 and REMD=8. The value of D sets the minimum time interval (D base clock cycles) between the leading edges of any two adjacent baud clock pulses, and the value of D+1 defines an extended time interval (D+1 base clock cycles) between selected pairs of adjacent baud clock pulses. In the FIGURE 6 example, the alternating sequence of minimum time intervals associated with count value D and extended time intervals associated with count value D + 1 conforms to the alternating sequence of 0s and 1s in the FIGURE 4 pulse position memory entry corresponding to REMD=8. In the example of FIGURE 6, the overall length of each bit interval is 8xD+8(D+1)=16D+8 base clock cycles. If D=3, for example, the bit interval length is 56 base clock cycles. This corresponds exactly to the baud rate given by: base clock rate/(K x D + REMD), for K=16, D=3 and REMD=8 which is: base clock rate/56. Although the example of FIGURE 6 shows 16 baud clock pulses per bit interval, it will be clear to workers in the art that the invention is more generally applicable to any baud clock having K pulses per symbol interval.

Returning now to the aforementioned example of a desired baud rate of 76,800, an oversampling factor of K=16 and a base clock rate of 1.8432 MHZ, the embodiment of FIGURE 2 yields D=1 and REMD=8. These values of D and REMD will yield a baud rate of: $1.8432 \times 10^6/16(1) + 8 = 76,800$. As another example, for a desired baud rate of 82,000, an oversampling factor of K=16 and a base clock rate of 1.8432 MHZ, the embodiment of FIGURE 2 yields D=1 and REMD=6. Thus, the resultant baud rate will be: $1.8432 \times 10^6/ 16(1) + 6 = 83,781.8$. This baud rate approximates the desired 82,000 rate significantly more closely than do the two closest possibilities yielded by the aforementioned prior art technique, namely 115,200 or 57,600.

FIGURE 7 illustrates another exemplary embodiment of the divisor generator of FIGURE 1. The embodiment of FIGURE 7 utilizes a look-up table 43 to produce the first and second divisor components D and REMD. As illustrated in more detail in FIGURE 8, the look-up table at 43 can index the possible pairs of values of D and REMD against respective baud rate ranges. For a given oversampling factor of K, there are K possible values of REMD (0 through K-1) for each possible value of D, because REMD is a modulo K remainder. Again using the example of a 1.8432 MHZ base clock rate and an oversampling factor of K=16: if D=1 and REMD=0, then the baud rate is 115,200; if D=1 and REMD=1, then the baud rate will be 108,423.5; and if D=1 and REMD=2, then the baud

rate will be 102,400. Thus, for example, Range 2 in the look-up table of FIGURE 8 could extend from a value halfway between 115,200 and 108,423.5 to a value halfway between 108,423.5 and 102,400. If the desired baud rate falls within Range 2, then the baud rate provided by the baud clock of FIGURE 1 will be 108,423.5, namely the baud rate associated with D=1 and REMD=1. The embodiment of FIGURE 7 includes a range classifier 41 which receives the desired baud rate and determines which of the ranges, Range 1, Range 2, etc. contains the desired baud rate. The range determined by the range classifier 41 is then input to the look-table 43, and corresponding values of the first and second divisor components D and REMD are output from the look-table.

It can be seen from FIGURES 1-8 that the composite nature of the enhanced granularity divisor D_{EG} advantageously results in K-1 intermediate baud rates available between each pair of adjacent baud rates produced using the conventional technique. For example, with a base clock frequency of 1.8432 MHZ and an oversampling factor of K=16, there are 15 intermediate baud rates (corresponding to REMD=1-15) available between 115,200 (D=1 and REMD=0) and 57,600 (D=2 and REMD=0).

FIGURE 9 diagrammatically illustrates pertinent portions of an exemplary data processing and communication system according to the present invention. In FIGURE 9, a data processing apparatus 61 is coupled at 69 for serial communication with one or more

external devices shown generally at 63. The data processing apparatus 61 includes data processing circuitry 65 and a serial communication interface 67 coupled to the data processing circuitry 65 for permitting serial communication between the data processing circuitry 65 and the external device(s) 63. In the exemplary embodiment illustrated in FIGURE 9, the serial communication interface 67 is a UART including the baud clock generator of FIGURE 1. The data processing circuitry 65 can perform any desired data processing operation(s) on data communicated between the data processing apparatus 61 and the external device(s) 63.

In one example, the data processing apparatus 61 is a microprocessor or digital signal processor, and the external device(s) at 63 can be any external device with which microprocessors and digital signal processors can communicate via the serial communication path shown diagrammatically at 69. For example, the external device(s) at 63 could be some other microprocessor(s) or digital signal processor(s). In some exemplary embodiments, the serial communication path at 69 is an RS-232 link.

In another example, the data processing apparatus 61 is the central processing unit (CPU) of a desktop or laptop computer, and the external device(s) at 63 is a keyboard, mouse, printer or any device that provides a man-machine interface to the central processing unit. In a further example, the data processing apparatus 61 and the external device(s) at 63

are modems coupled via serial interface(s) to permit communication between further devices that are not explicitly shown in FIGURE 9. In a still further example, the data processing apparatus 61 is a radiotelephone and the external device 63 is a laptop or desktop computer.

FIGURE 10 illustrates exemplary operations which can be performed by the embodiments of FIGURES 1-9. It is first determined at 71 whether a new baud rate is desired. If so, then D and REMD are obtained at 73. Thereafter D and REMD are used at 75 to produce the baud clock from the base clock.

FIGURE 11 diagrammatically illustrates an example of a baud rate detector with which an external device such as shown at 63 in FIGURE 9 can determine what baud rate is being used by the communication interface 67 of FIGURE 9. In the example of FIGURE 11, a counter counts the number of base clock cycles that occur during the START bit conventionally associated with the incoming data. Assuming in this example an oversampling factor K = 16, the 4 ($2^4 = 16$) least significant bits of the output count value will represent REMD (REMD = 1 in this example) and the remaining most significant bits represent D (D = 1 in this example). These D and REMD values can then be applied to the clock divider 13 of FIGURES 1 and 3, which can then produce the baud clock needed to sample the incoming data. Thus, the baud rate detector of FIGURE 11 is another example of a divisor generator for use with the clock divider 13.

It should be evident from the foregoing that use of the enhanced granularity divisor including first and second divisor components according to the present inventions provides a significantly larger number of available baud rates given a fixed base clock frequency. This permits the desired baud rate to be approximated much more closely than in the prior art, particularly at relatively high baud rates. This closer approximation can advantageously be provided without requiring an increase in the base clock frequency.

Although exemplary embodiments of the invention are described above in detail, this does not limit the scope of the invention, which can be practiced in a variety of embodiments.